M-SEC-905

Offensive Security and exploitation Pentest Report

Marian Bret

Ka Po Chau

Nejma Belkhanfar

Table of Contents

Methodology

Work methodology :

We tried to keep the same methodology for every single information gathering or exploitations we did on the scope 10.10.10.0/24.

The recommended methodology about pentesting is this pattern : Footprinting > Network scanning > Enumeration > Exploitation.

First part of the work was to scan the network scope to find the first useful information : the hosts that were up and ports that were open on those hosts.

As we spotted some DNS open ports, we decided to scan them to gather as much information as possible (logs available in 'log-10.10.10.10-53' and 'log-10.10.10.11-53'). Those scans allowed us to get some useful information about hosts and PTR (which are "reverse DNS" that give the domain name associated with an IP). Those informations were crucial for deeper scans and possible exploitations.

After getting the whole list of connected hosts and open ports that belong to those hosts, we continued to follow the methodology of pentesting which led to some deeper network scans on a specific host and port (the list is available on 'scan-logs'). We usually used Kali Linux's built-in scanners softwares such as Nmap (https://nmap.org/) or Dirb for http ports as an example. Those scanners have multiple parameters and flags that allow them to do a specific and precise scan to gather information such as services that are running currently on the network with their specific versions, actual configurations, responses of the server…

During those scans, we used some built-in scripts in scanning softwares such as nmap-script-engine (https://nmap.org/man/fr/man-nse.html) or Metasploit scanners that can led to leak of some impacting informations on the victim host (either with the -sC Nmap flag that launch the default script for a port or either precise scripts, using the —script= flag on Nmap).

This scanning work gave us a good overview of the network scope and possible entries vectors to look for possible vulnerabilities deeper.

Keeping on following the pentest methodology, we tried to enumerate as much information as possible on a host/port we found open earlier. To achieve this, we used some softwares and databases such as searchsploit, exploit-db, CVE-details and built-in enumeration scripts (such as SMB enumeration or enum4linux for example).

With all those scanning and enumerating information, we can try to find the good attack vector on a specific host. At this point, we got two options, either the host/port is vulnerable and we managed to exploit it to find some very impacting information or established a root connexion (or vertical/horizontal privilege escalation) or either we found some vulnerabilities and security flaws that haven't been exploited but can led to future exploits. As long as we didn't fulfil in breaking the protections, we continued to follow the methodology in a way to do deeper scans or enumerate more useful information about the system. In fact, we understood that a machine or service we tried to break that led to nothing doesn't mean that the service isn't vulnerable or dangerous for the system (as an example, a service can leak no information but can be vulnerable for Dos attack or else).

The structure we used to store the information we gathered and our exploits is available on the project Github (https://github.com/marianbret/offensive_security_tek5) with this actual PDF report and some logs that follow the same pattern : log-host-port.txt (for example: log-10.10.10.10-22.txt). Those logs contain the scanning information, the enumerations and exploitation.

Scanning

Scanning the project scope:

The project scope is : 10.10.10.0/24

The first machine of the scope is : 10.10.10.1

The last machine of the scope is : 10.10.10.254

The scope scanning logs are available on Github in the ‘scans-logs.txt’ file.

Scanning SSH ports:

The SSH ports scanning are available on Github in the ‘scans-ssh-logs.txt’ file.